Invariance Priors for Bayesian Feed Forward Neural Networks

Udo v. Toussaint, S. Gori and V. Dose
Max-Planck-Institut für Plasmaphysik, EURATOM Association, Boltzmannstr. 2, D-85748 Garching, Germany

Neural Networks are famous for their advantageous flexibility for problems when there is insufficient knowledge to set up a proper model. On the other hand this flexibility can cause over-fitting and can hamper the generalization properties of neural networks. Many approaches to regularize NN have been suggested but most of them based on ad-hoc arguments. Employing the principle of transformation invariance we derive a general prior in accordance with the Bayesian probability theory for feedforward networks. An optimal network is determined by Bayesian model comparison verifying the applicability of this approach. Additionally the presented prior affords cell pruning.

PACS numbers: 07.05.Kf, 07.05.Mh, 84.35.+i
Published in Neural Networks: Received 8 December 2003
Accepted for publication 4 January 2006
Published online 31 March 2006
Neural Networks Vol.**19**, Issue 10 (2006) p. 1550-1557

Symbols:

H	a network structure
$\vec{w}$	a network weight vector
$\vec{D}$	a data vector
$\vec{b}$	a bias vector
$\mathbf{H}$	Hessian matrix
E	number of parameters of a network
K	number of neurons in a hidden layer
k	indices of hidden neurons
N	number of neuron inputs
i	indices of neuron inputs
q_{ij}	probability of pixel ij being 1
r_0	lower cut-off for prior term
$T_\epsilon(\vec{w})$	infinitesimal transformation of $\vec{w}$
$p(\vec{w}\|I)$	probability density of $\vec{w}$
$g(.)$	activation function of a network
λ	scale parameter

I. INTRODUCTION

Neural networks (NN) have attracted a lot of interest. One reason is that they provide a useful tool for function approximation, classification and density estimation. Furthermore trained NN have excellent response times. This is crucial for many real-time applications. However, the problem of selecting the appropriate structure of the network (ie number of hidden neurons) remains a critical issue for NN. The number of neurons controls the complexity, and hence the generalization ability of a NN. A NN with too many neurons gives poor generalization since it is too flexible and fits the noise in the data. On the other hand, a NN with too few neurons also yields poor predictions of new data, since the model does not incorporate all available information. With standard neural networks techniques, the means for determining the appropriate number of neurons are rather arbitrary. In the Bayesian approach, these issues can be handled in a consistent way. This approach for NN is well established since the work of MacKay [1] and Neal [2] and has been reviewed in Bishop [3] and Lampinen & Vehtari [4]. One of the key observations was that the conventional error function of NN can be interpreted as minus the log-likelihood, giving a probabilistic interpretation to the NN optimization process. Similarly the Tikhonov-style regularizer is reinterpreted in terms of a log prior probability of the parameters. The most often used quadratic regularizer on the weights $E(w) = 1/2 \sum_i w_i^2$ then conveniently corresponds to a Gaussian prior distribution with mean zero. Having made these assignments Bayesian probability theory (BPT) can be used for model selection and determination of the correct complexity of a NN: Various models of different complexity are optimized and the resulting NN are compared by the evidence given the data sample. Since the evidence takes into account the quality of the fit and the volume of the parameter space more complex models are penalized if the additional parameters give only negligible improvements of the fit [**?**]. There are some differences in the technical details (eg use of Laplace approximation or MCMC) but the general procedure is straightforward and given by the BPT.

However, it should be noted that the use of a gaussian prior was primarily motivated by the widespread use of the quadratic regularizer in the conventional approach. This choice was not based on any first principles nor is the

quadratic prior unique [3, 5, 7]. Unfortunately the success of Bayesian NN has shifted the search for appropriate priors for NN out of focus. This paper carefully inspects the prior information contained in the structure of NN and uses the principle of transformation invariance [8] and the the principle of maximum entropy [9] to derive informative prior probability density distributions. The importance of invariance principles for prior selection have also been emphasized by Williams [6]. Starting from the basic requirement of consistency for a statistical model that it should be independent of the labeling of variables he derived a permutation invariant parametrization for a neural network covariance model. For this model he further investigated which restrictions are imposed to regularizers of the form

$$p\left(\vec{w}|p,\gamma,I\right)=\left(\left(\sum_i |w_i|^p\right)^{1/p}\right)^{-\gamma},\ \gamma>0 \tag{1}$$

by consistency requirements - i.e. to which extent the chosen combination of parametrization of the covariance model and prior is invariant under general linear transformations of the variables. One of the results was that the regularizer Eq. (1) is invariant under rotations for $p=2$, a property which does not hold for $p=1$.
Our approach is even more fundamental. Instead of considering the invariance properties of various types of regularizers under variable transformations we start with the requirements of rotation and translation invariance of the a-priori network response and show that these requirements are sufficient to derive a prior function for the network weights.

II. THE BAYESIAN APPROACH

The tools needed for the analysis are provided by the Bayesian probability theory (BPT). The BPT rests on the application of two rules. The first is the product rule. Given a probability $P\left(D,w|H,I\right)$ depending on two or more variables conditional on a model H (eg a neural network) and additional information I, the product rule allows to expand $P\left(D,w|H,I\right)$ into simpler densities depending only on either w or D as a variable

$$P\left(D,w|H,I\right)=P\left(w|H,I\right)P\left(D|w,H,I\right)=P\left(D|H,I\right)P\left(w|D,H,I\right). \tag{2}$$

Comparison of the two alternative expansions yields Bayes theorem

$$P\left(w|D,H,I\right)=\frac{P\left(w|H,I\right)P\left(D|w,H,I\right)}{P\left(D|H,I\right)}. \tag{3}$$

In order to interpret Eq. (3) D is associated with data providing information on the parameters (eg network weights) w. Bayes theorem relates the posterior probability density function (pdf) $P\left(w|D,H,I\right)$ to the likelihood pdf $P\left(D|w,H,I\right)$ and the prior pdf $P\left(w|H,I\right)$. The likelihood $P\left(D|w,H,I\right)$ is the probability that we measure the data D *assuming* w is known. The prior probability $P\left(w|H,I\right)$ is the probability that we attach to a particular value of w before the data D is taken into account. The denominator $P\left(D|H,I\right)$ in Bayes theorem is called the evidence. The evidence can be calculated using the second, the so-called marginalization rule of BPT

$$P\left(D|H,I\right)=\int dw\,P\left(w,D|H,I\right)=\int dw\,P\left(w|H,I\right)P\left(D|w,H,I\right) \tag{4}$$

and is the normalization in Bayes theorem. Furthermore $P\left(D|H,I\right)$ represents the probability of the data given a hypothesis H regardless of the actual (optimized) numerical parameter values. The evidence is crucial in ranking different models based on the same set of data since the posterior probability for a model H_i is

$$P\left(H_i|D,I\right)\propto P\left(D|H_i,I\right)P\left(H_i|I\right). \tag{5}$$

The second term $P\left(H_i|I\right)$ is the subjective prior over our hypothesis space expressing how plausible we thought the alternative models were before the data arrived. Assigning equal prior probabilities to the alternative models, the models H_i are ordered by the evidence.

A more detailed introduction to BPT can be found in the literature [10–12].

III. HYPERPLANE PRIORS

To obtain the posterior distribution of the weights and the evidence for the different models we need to specify the likelihood of the data and the prior distribution for the parameters. The likelihood is model dependent and usually

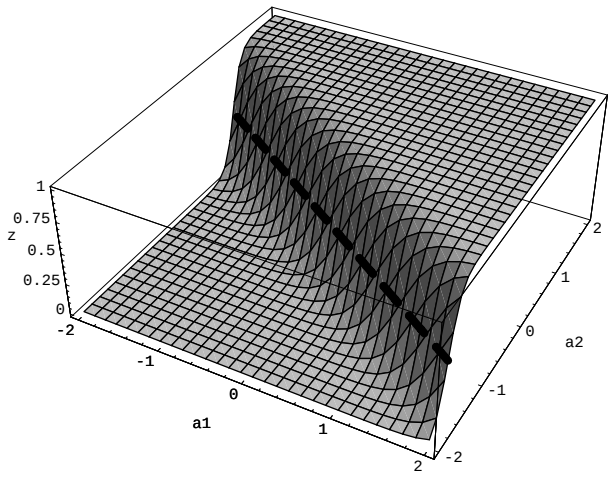

FIG. 1: One dimensional decision boundary of a neuron with two inputs ($N = 2$) indicated with a line. The extent of the transition area depends on the value of b in Eq. 7.

there is a fair amount of knowledge which one to select. The correct choice of the prior is not so obvious especially for 'non-parametric' models like NN. If we consider a typical neuron of a neural network with N incoming connections with activations $a_i, i = 1...N$ and weights $\tilde{w}_i, i = 1...N$ then the output z is given by

$$z = g\left(b + \sum_{i=1}^{N} a_i \tilde{w}_i\right), \tag{6}$$

where b denotes the bias and g is the activation function. Assuming one of the standard activation functions (Heaviside function, tanh, or logistic sigmoid) Eq. 6 can be considered as linear discriminant function since the decision boundary which it generates is linear, as a consequence of the monotonic nature of $g(.)$. The decision boundary

$$b + \sum_{i=1}^{N} a_i \tilde{w}_i = b\left(1 + a_1 w_1 + a_2 w_2 + ... + a_N w_N\right) = 0 \tag{7}$$

corresponds to an $(N-1)$-dimensional hyperplane in N-dimensional w-space. For the case of a two-dimensional input space, N=2, the decision boundary is a straight line, as shown in figure 1. A priori we should not favor any orientation or position of this decision boundary and this must be reflected in the prior [8]. Therefore we require that the prior is invariant under rotations and translations of the coordinate system

$$p\left(\vec{w}\right) dw_1 dw_2 ... dw_N = p\left(\vec{w'}\right) dw_1' dw_2' ... dw_N' \tag{8}$$

where $p\left(\vec{w}\right) d\vec{w}$ is an element of probability mass whose value must be independent from the system of coordinates used to evaluate it. Using the Jacobian of the transformation $\vec{w} \rightarrow \vec{w'}$ (following closely Dose [13]) we obtain the equation

$$p\left(\vec{w}\right) = p\left(\vec{w'}\right) \det\left(\frac{\partial w_i'}{\partial w_k}\right). \tag{9}$$

Since any finite transformation can be constructed from a sequence of infinitesimal transformations it is sufficient to consider a single infinitesimal transformation $\vec{w'} = T_\epsilon\left(\vec{w}\right)$. Then Eq. (9) can be rewritten as

$$p\left(\vec{w}\right) = p\left(T_\epsilon\left(\vec{w}\right)\right) \det\left(\frac{\partial T_\epsilon\left(\vec{w}\right)}{\partial w_k}\right). \tag{10}$$

After differentiating with respect to ϵ we obtain the functional equation following from the requirement of transformation invariance:

$$\frac{\partial}{\partial \epsilon}\left[p\left(T_\epsilon\left(\vec{w}\right)\right) \det\left(\frac{\partial T_\epsilon\left(\vec{w}\right)}{\partial w_k}\right)\right]\Bigg|_{\epsilon=0} = 0. \tag{11}$$

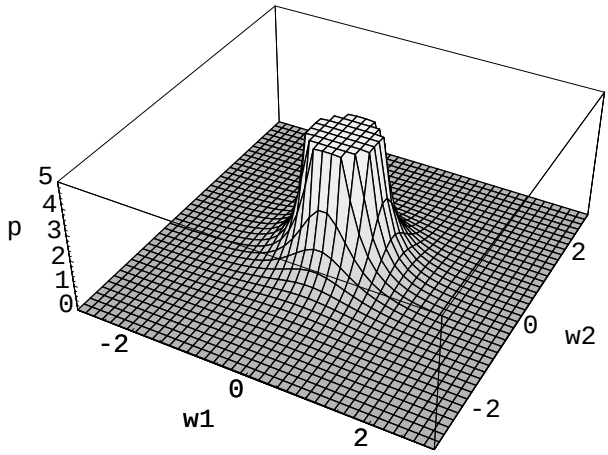

FIG. 2: Prior distribution of the weights (Eq. 13) for $N = 2$. Note the radial symmetry and that smaller weights are strongly preferred compared to larger ones. The prior distribution is not defined for $\| \vec{w} \|_2 < r_0$. Details about the choice of r_0 are given in the text.

This equation (with a typo in Dose [13] corrected) has to be fulfilled for the general rotation and translation in N-dimensional space.

A. Weight Prior

For the hyperplane equation

$$1 + a_1 w_1 + a_2 w_2 + \ldots + a_N w_N = 0 \tag{12}$$

this calculation yields the normalized prior (see Appendix):

$$p\left(w_1, \ldots, w_N | I\right) = \frac{\Gamma\left(N/2\right) r_0}{2\pi^{N/2}} \frac{1}{\left(w_1^2 + \ldots + w_N^2\right)^{\frac{N+1}{2}}}, \quad ||\vec{w}||_2 \geq r_0 > 0 \tag{13}$$

with a lower limit r_0 for the norm of the weight-vector. A suitable choice for r_0 can be derived as follows: Let $(a_1^*, ..., a_N^*)$ be the point closest to the origin through which the hyperplane passes. Then the lowest possible value of $r^2 = \sum_{i=1}^{K} w_i^2$ is given by the minimum of the Lagrangian optimization problem

$$\Phi = 2\lambda \left(1 + \sum_{i=1}^{N} w_i a_i^*\right) + \sum_{i=1}^{N} w_i^2 \tag{14}$$

which yields

$$r^2 = \frac{1}{\sum_{i=1}^{N} a_i^{*2}}. \tag{15}$$

Therefore, assuming scaled input data (eg. $a_i \in [0, 1], \forall i \in \{1, ..., N\}$) an estimate is obtained by $r_0^2 = \left(N \cdot \max |a_i|\right)^{-1}$. The decision boundary of a neuron is shifted farther away from the origin when r reduces towards r_0. The output of this neuron is then constant (within numerical precision) for the input data. The prior distribution (13) is visualized for $N = 2$ in Fig. (2). Being normalizable this prior can be used for model comparison, where proper priors are essential.

B. Bias prior

To assign a prior for the bias parameters $p\left(\vec{b}|I\right)$ we use the maximum entropy principle, using the maximal slope of the decision boundary as relevant testable information. As can be seen from Eq. (7) $\vec{b}$ determines the width

of the transition or 'the sharpness' of the separation. Assuming for definiteness the logistic activation function $g(x) = (1 + \exp(-x))^{-1}$ the slope of the decision boundary is given by the gradient

$$\nabla g = \frac{g}{(1+g)^2} b \begin{pmatrix} w_1 \\ \vdots \\ w_N \end{pmatrix} \tag{16}$$

and is therefore perpendicular to the orientation of the hyperplane. The maximum value of the gradient $|\nabla g|^2 = \frac{1}{16} b^2 \sum_{i=1}^N w_i^2$ is taken for the hyperplane points fulfilling $1 + \sum_{i=1}^N w_i a_i = 0$. The maximum entropy principle assigns this measurable quantity the normalized prior

$$p(b|\vec{w}, \lambda, I) = \sqrt{\frac{\lambda \sum_{i=1}^N w_i^2}{2\pi}} \exp\left(-\frac{\lambda}{2} b^2 \sum_{i=1}^N w_i^2\right), \tag{17}$$

where we had to introduce the hyperparameter λ as a scale parameter, reflecting the uncertainty of the magnitude of the slope before we have any information about the data. Using again the transformation invariance principle for this scale parameter we obtain Jeffreys' prior $p(\lambda|I) \propto 1/\lambda$. It should be pointed out that Jeffreys' prior is not normalizable. It can, however, be considered as a limiting distribution of a sequence of proper gamma priors

$$p(\lambda|I) \propto 1/\lambda = \lim_{c \to 0} \frac{c^c}{\Gamma(c)} \lambda^{c-1} \exp(-c\lambda). \tag{18}$$

Setting $B = \sum_{k=1}^K b_k^2 \sum_{i=1}^N w_{ik}^2$, with w_{ik} being the weight connecting input i to neuron k we generalize to K neurons in a hidden layer. Then, using BPT for marginalizing the nuisance parameter λ we can write

$$\begin{aligned} p\left(\vec{b}|\vec{w}, I\right) &= \lim_{c \to 0} \frac{c^c}{\Gamma(c)} \int_0^\infty d\lambda \left(\prod_{k=1}^K \sqrt{\frac{1}{2\pi} \sum_{i=1}^N w_{ik}^2}\right) \lambda^{\frac{K}{2}+c-1} \exp\left(-c\lambda - \frac{\lambda}{2} B\right) \\ &= \lim_{c \to 0} \frac{c^c}{\Gamma(c)} \left(\prod_{k=1}^K \sqrt{\frac{1}{2\pi} \sum_{i=1}^N w_{ik}^2}\right) \underbrace{\frac{\Gamma\left(\frac{K}{2}+c\right)}{\left(c + \frac{1}{2}B\right)^{\frac{K}{2}+c}}}_{\Psi}. \end{aligned} \tag{19}$$

Ψ is only of interest in the region of maximum likelihood. There B, the weighted sum of all squared network weights is much larger than c and also $K \geq 1 \gg c$. Hence later employing the Laplace approximation at the maximum we can take $c = 0$ in Ψ. The normalization factor $c^c/\Gamma(c)$ is the same in all considered models and is therefore irrelevant for model comparison. Combining Eq. (19) with the prior for the weights (13) we finally obtain as prior for the weights and biases of a single layer

$$p\left(\vec{b}, \vec{w}|I\right) = \left(\frac{\Gamma\left(\frac{N}{2}\right) r_0}{\pi^{\frac{N+1}{2}} 2}\right)^K \underbrace{\frac{\Gamma\left(\frac{K}{2}\right)}{\prod_{k=1}^K \left[\left(\sum_{i=1}^N w_{ik}^2\right)^{\frac{N}{2}}\right]}}_{\phi} \underbrace{\frac{1}{\left(\sum_{k=1}^K b_k^2 \sum_{i=1}^N w_{ik}^2\right)^{\frac{K}{2}}}}_{\tau}. \tag{20}$$

C. Extension to Multilayer Neural Networks

Taking advantage of the fact that the invariance argument applies to the network weights of each subsequent layer individually if we consider the preceeding layers as 'black box' the prior for a multilayer neural network is given by the product of the priors (Eq. (20)) of the individual layers. Please note that we neglect information about the network structure considering preceeding layers as 'black boxes'.

D. Pruning Properties

Responsible for the pruning properties of the prior is mainly the second term in Eq. (20), ϕ. For a trained NN, the sensitivity of the data misfit to a given weight has to counterbalance the sensitivity of the prior (focusing on the

relevant term ϕ):

$$\frac{\partial \log(\phi)}{\partial w_{ab}} \propto -\frac{w_{ab}}{\sum_{i=1}^{N} w_{ib}^2}. \tag{21}$$

The dependence of the sensitivity on **all** weights on the incoming connections of a cell is the underlying reason for the pruning of cells instead of single weights. Additionally, a hyperplane positioned far outside the data space (and therefore $\sum_{i=1}^{N} w_{ib}^2$ being small) needs a higher sensitivity of the data misfit to its weights compared to a hyperplane crossing the data space. Finally, the third term in (20) favors smooth decision boundaries, being the only prior term dependent on $\vec{b}$. Rewritten in terms of conventional network weights, $\tau = \| \tilde{\vec{w}} \|_2^{-K}$, it becomes obvious that this term assigns a high probability to layers with small weights.

IV. MODEL SELECTION

Assuming prior (20) and likelihood (eg Eq.(24)) being specified we can use the Bayesian probability theory for model comparison. Please note that Eq. (20) does not contain any hyperparameters - therefore rendering an iterative estimation procedure [1] for the optimal hyperparameters unnecessary. The Bayesian approach allows model comparison of networks with different numbers of hidden units without the need for separate training and validation data - making all the information in the data available for the training of the NN. According to Eq. (4) the evidence for a neural network is obtained by marginalizing over all network parameters $\vec{b}$ and $\vec{w}$:

$$p\left(H|\vec{D},I\right) = \int_{-\infty}^{\infty} d\vec{b}\, d\vec{w}\, \underbrace{p\left(\vec{D}|\vec{b},\vec{w},H,I\right) p\left(\vec{b},\vec{w}|I\right)}_{\exp(-\phi)}. \tag{22}$$

To evaluate this integral analytically we will employ the standard Laplace approximation, expanding ϕ to second order around its maximum at $\left(\vec{b}^*,\vec{w}^*\right)$. Then the evidence is given by

$$p\left(H|\vec{D},I\right) \approx p\left(\vec{D}|\vec{b}^*,\vec{w}^*,H,I\right) p\left(\vec{b}^*,\vec{w}^*|I\right) \frac{(2\pi)^{E/2}}{\sqrt{\det(\mathbf{H})}} \tag{23}$$

where $\mathbf{H}$ is the Hessian of dimension $E \times E$ at $\left(\vec{b}^*,\vec{w}^*\right)$, with E being the number of parameters.

V. EXAMPLES

To illustrate the properties of the prior (20) we use an 2-dimensional example shown in Fig. 3. The figure shows a simulated data set designed to match binary images from phase shifting speckle interferometry of microscopically rough surfaces [14]. The clear separation of the interferometric pattern from the noise is a necessary prerequisite for the subsequent surface reconstruction from the interferogram. The likelihood for this problem is a binomial one, given by

$$p\left(\vec{D}|\vec{q},I\right) = \prod_{i,j}^{N_x,N_y} q_{ij}^{D_{ij}} \left(1-q_{ij}\right)^{(1-D_{ij})} \tag{24}$$

where q_{ij} is the probability for pixel D_{ij} being 1. The discrete domain of the data together with the poor signal to noise ratio is one reason for the difficulties of many other techniques to extract the underlying fringe pattern.

In the simulation we applied a feedforward network with two input neurons, one layer of hidden units and a single output with a sigmoid activation function to restrict the output values to the appropriate interval $[0,1]$. Adjacent layers had all-to-all connections. The weights of the connections from the hidden layer to the output neuron were fixed. This seemed appropriate for the test case (Fig.3) with its fringe-no fringe structure (Subsequent computations, however revealed a faster convergence if also those weights were optimized). A set of 100 neural networks, each with randomly chosen weights, was trained with the data visualized in Fig. 3 with the number of hidden neurons ranging from 1 to 50. The first iterations of the optimization algorithm [15] were performed without prior allowing the net to find interesting structures. Then the optimization was run with prior until a minimum was found. Contrary to

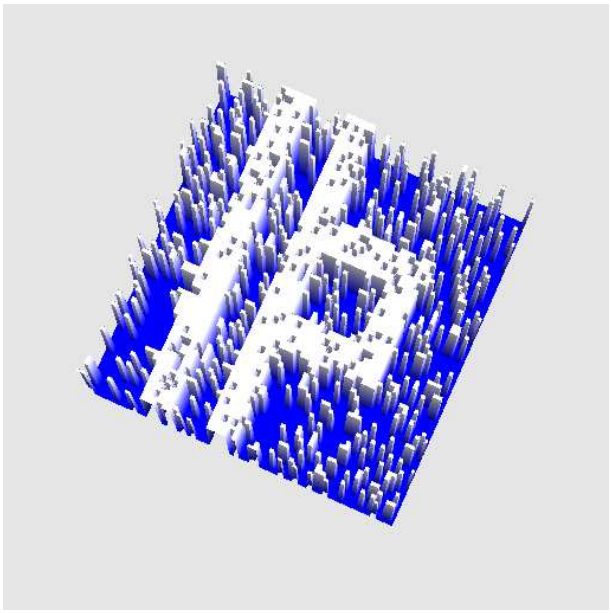

FIG. 3: Simulated binary (0,1)-speckle image with 64×64-pixels. The actual noise level is set to 0.12.

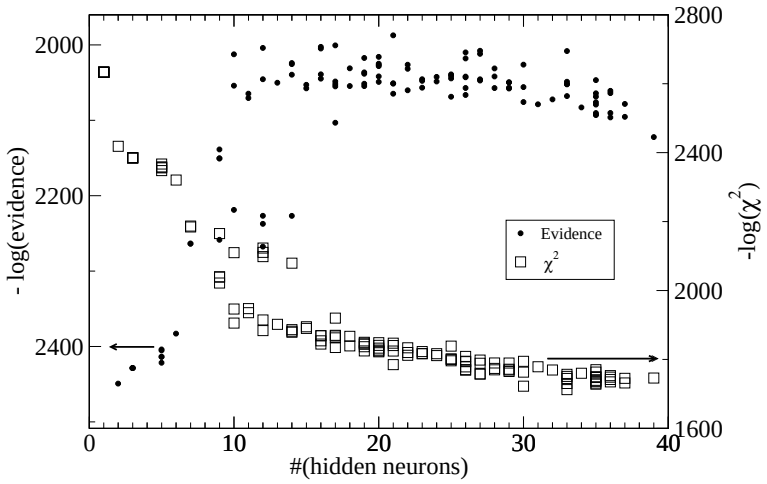

FIG. 4: The error of the neural network on the data set is monotonically decreasing with increasing number of neurons. The evidence exhibits a strong increase left of the minimum indicating that the neural net has not enough hidden neurons to fit essential structures. The slow decrease to the right side shows that Ockham's razor penalizing the increased model complexity is not compensated by better fits.

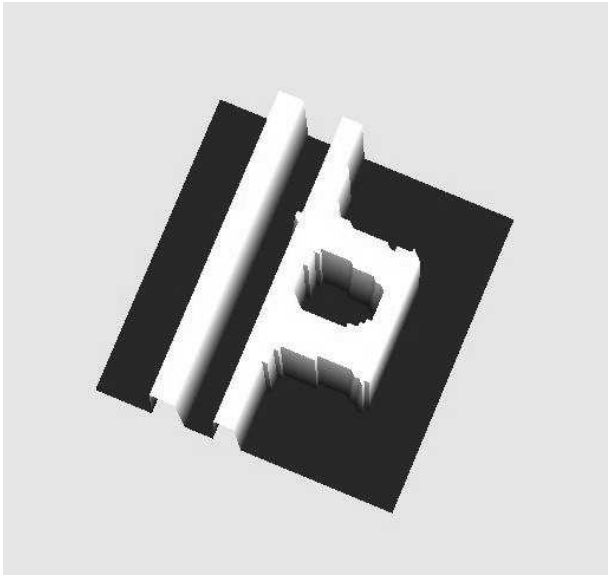

FIG. 5: The result of the neural network with the highest evidence using a simple threshold for classification. Note the absence of any spurious peaks in areas not connected to the structure.

other findings (eg. in [16, 17]) the analytical Hessian was positive definite in all cases thus not requiring any cutoff of the eigenvalues for taking the square root of the determinant. If a neuron had reached the lower cut-off limit $\| \vec{w} \|_2 = r_0$ after the optimization, this neuron was pruned and a possible offset was added to the bias of the next layer. Subsequently the pruned network was again optimized. All NN with initially between 40 and 50 neurons were pruned to less than 40 neurons. The results are shown in Fig. 4. The misfit is monotonically decreasing with increasing model complexity. For low model complexity the evidence is dominated by the error of the fit as can be seen by the inverse behaviour of the evidence and the likelihood. This reflects the incapability of the neural net to fit essential data structures. If, on the other hand, the model complexity is too high then the larger parameter space consumption cannot be counterbalanced by the only slightly better fit (Ockham's razor). The slow decrease of the evidence despite the increasing likelihood is a direct indication for this. The neural network with the highest evidence has 21 neurons but the evidence exhibits an extended maximum for networks with 10 to 25 hidden neurons.

It seems wasteful to train 100 networks and use only the best one. In the Bayesian framework, it is natural to consider ensembles of networks. Forming a committee of all networks is usually straightforward by weighting the predictions according to the evidence [17]. In our case, despite the fact that the evidence is not sharply peaked, the absolute scale of the evidence prevents a noticeable contribution of more than the few best networks to the presented result. Taking a different point of view those 100 trained networks resemble the result of a simple Monte Carlo simulation, each obtained result being a local optimum and therefore reflecting the multimodal evidence surface. The output of this committee is given in Fig. 5 where each pixel was assigned the binary value with the higher probability. The separation of noise and the underlying fringe pattern is excellent. There are only minor deviations from the true structure near the edges of the fringes.

In a further example the noise level was much higher. The Fig.6 shows two simulated 256x256 pixel data sets with different geometric structure. The upper row shows the undistorted test data. The originals were degraded using a binomial distribution with p(white pixel| white)=p(black pixel| black)=0.72 (Fig.6, middle row). Instead of only a few pixels being distorted in Fig.3 the binomial noise of $p = 0.28$ renders the underlying structure of the test examples displayed in Fig.6 on a small scale almost invisible. The same network structure as in the previous example was used. A set of 20 networks, each with randomly chosen weights, was trained with the data visualized in Fig.6, middle row. The number of hidden neurons were ranging from 1 to 30 and the same iterative optimization procedure as before was applied. All NN with initially between 20 and 30 neurons were pruned to less than 20 neurons. The output of the networks with the highest evidence is given in Fig.6, lower row, where each pixel has been assigned the more likely binary value. The noise filtering capabilities are very good. There are minor deviations from the true structure near the edges of the fringe for the left test example. The example on the right side is accurately reconstructed.

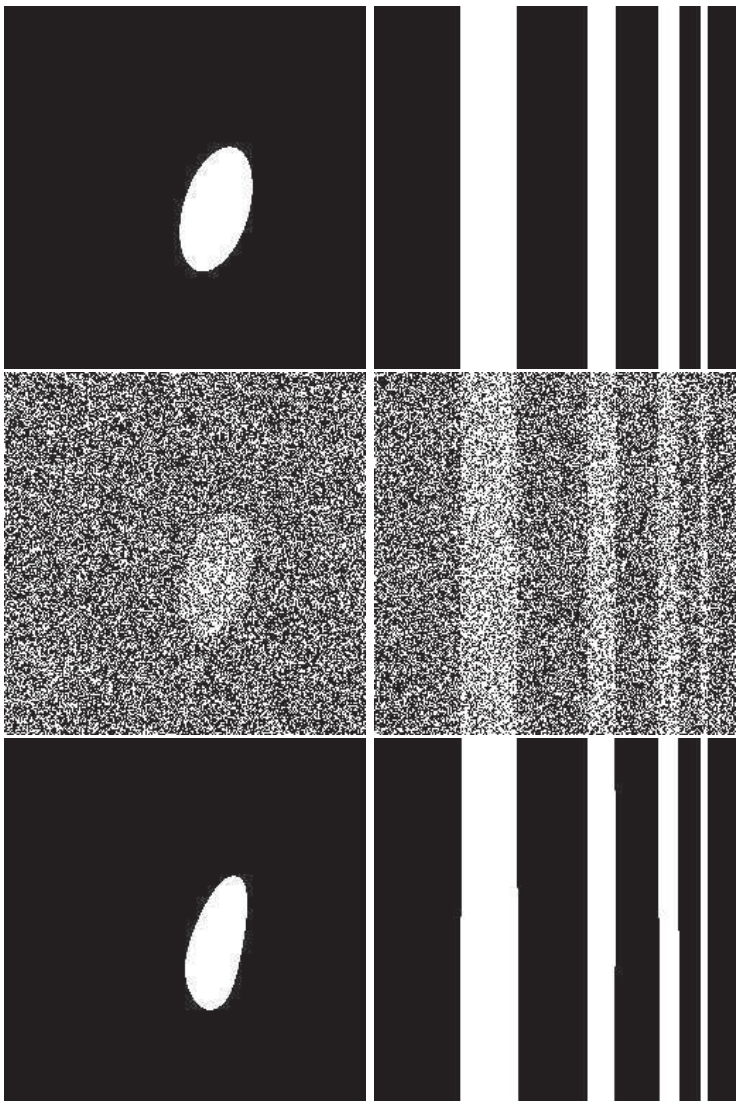

FIG. 6: Upper row: undistorted binary images; Middle row: Noisy binary test images using a binomial probability distribution with p(black pixel— black)=p(white pixel— white)=0.72;Lower row: Result of the Bayesian neural network with the highest evidence.

VI. CONCLUSIONS AND OUTLOOK

We have proposed a hyperplane prior for the weights of a class of feedforward neural networks based on the mandatory requirement of transformation invariance which is violated by other regularization methods or priors. Additionally, the structure of the presented prior favors pruning of complete neurons instead of only individual weights, which is of interest from a practical point of view. A Bayesian model comparison utilizing this invariant prior has been performed. Our experimental results validate the effectiveness of this approach. In several 2-dimensional test examples using a binomial likelihood we could demonstrate that overfitting is effectively suppressed by cell pruning and that the evidence analysis using this prior gives excellent results. Currently the approach is under investigation for an automated analysis of speckle measurements of plasma facing materials [18] to determine surface changes.

APPENDIX A: DERIVATION OF THE INVARIANT HYPERPLANE PRIOR

For the general case of an (n-1) dimensional plane in n-dimensional space the hyperplane in the original and transformed coordinate system are

$$1 + a_1 w_1 + a_2 w_2 + \ldots + a_N w_N = 0, \tag{A1}$$
$$1 + a'_1 w'_1 + a'_2 w'_2 + \ldots + a'_N w'_N = 0. \tag{A2}$$

The general rotation in n-dimensional space may be expressed as a sequence of two-axis (j,k) rotations [19]. There exist $\binom{n}{2}$ such rotations eg. one for n=2, three for n=3, six for n=4 etc. Now we perform one such rotation: $a'_i = a_i$ for all i not equal to either j or k and

$$\begin{aligned} a'_j &= a_j \cos\phi - a_k \sin\phi \approx a_j - \epsilon a_k, \\ a'_k &= a_j \sin\phi + a_k \cos\phi \approx \epsilon a_j + a_k. \end{aligned} \tag{A3}$$

Substituting the primed coordinates into A2 and collecting coefficients of a_i yields the implied transformation

$$\begin{aligned} w'_j &= w_j - \epsilon w_k \\ w'_k &= w_k + \epsilon w_j \text{ and} \\ w'_i &= w_i \forall i \neq j,k. \end{aligned}$$

The Jacobian of the transformation is

$$\frac{\partial\left(a'_1,\ldots,a'_n\right)}{\partial\left(a_1,\ldots,a_n\right)} = \begin{vmatrix} 1 & & & & & & \\ & \ddots & & & & 0 & \\ & & 1 & \cdots & -\epsilon & & \\ & & & \ddots & & & \\ & & \vdots & & 1 & \vdots & \\ & & & & & \ddots & \\ & & \epsilon & \cdots & & 1 & \\ & 0 & & & & & \ddots \\ & & & & & & 1 \end{vmatrix} = 1 + \epsilon^2 \tag{A4}$$

and the transformation invariance functional equation 11 leads to the partial differential equation

$$w_k \frac{\partial p}{\partial w_j} = w_j \frac{\partial p}{\partial w_k} \tag{A5}$$

from which follows that p can only be a function of $w_j^2 + w_k^2$. Since this must hold for any pair j,k we conclude that

$$p\left(w_1, w_2, \ldots, w_n\right) = p\left(w_1^2 + w_2^2 + \ldots + w_n^2\right). \tag{A6}$$

Now p must also satisfy invariance under the n possible translations

$$a'_k = a_k + \epsilon, \text{ and } a'_i = a_i \; \forall \; i \neq k. \tag{A7}$$

We substitute A7 into A2 and obtain

$$\sum_i \frac{w_i'}{1+\epsilon w_k'} a_i + 1 = 0 \tag{A8}$$

from which we read the implied transformation

$$w_i = \frac{w_i'}{1+\epsilon w_k'} \;\longrightarrow w_i' = w_i + \epsilon w_i w_k \tag{A9}$$

with the Jacobian

$$\frac{\partial\left(w_1', \ldots, w_n'\right)}{\partial\left(w_1, \ldots, w_n\right)} = \begin{vmatrix} 1+\epsilon w_k & & & & \\ & \ddots & & & \\ & & 1+2\epsilon w_k & & 0 \\ & & & \ddots & \\ & & 0 & & 1+\epsilon w_k \end{vmatrix} = 1 + (n+1)\epsilon w_k. \tag{A10}$$

The unknown function p must therefore obey the second set of partial differential equations

$$w_1 w_k \frac{\partial p}{\partial w_1} + \ldots + w_k^2 \frac{\partial p}{\partial w_k} + \ldots + w_n w_k \frac{\partial p}{\partial w_k} + w_k\left(n+1\right) p = 0. \tag{A11}$$

Equation A11 may be simplified if we require $w_k \neq 0$ to

$$\sum_i w_i \frac{\partial p}{\partial w_i} + (n+1)p = 0. \tag{A12}$$

Translation of other coordinates, say a_j, leads to exactly the same differential equation A12 however with the side condition $w_j \neq 0$. We now use the previous result A6 that p can only be a function of $\rho = w_1^2 + w_2^2 + \ldots + w_n^2$. Then the derivative with respect to w_i is given by $\partial p/\partial w_i = 2w_i\, dp/d\rho$ and we arrive at

$$2\sum_i w_i^2 \frac{dp}{d\rho} + \left(n+1\right) p = 0 \tag{A13}$$

which is readily integrated to yield

$$p\left(w_1, \ldots, w_n\right) = \frac{1}{\left[w_1^2 + \ldots + w_n^2\right]^{\frac{n+1}{2}}}\,, |w_i| > 0. \tag{A14}$$

The condition $|w_i| > 0$ in A14 means of course that the distribution is normalizable. Denote by $r^2 = w_1^2 + w_2^2 + \ldots + w_n^2$ and by r_0 the minimum value of r which we allow. The normalization integral is then

$$Z = \int d\Omega_n \int_{r_0}^{\infty} \frac{dr}{r^2} = \frac{2\pi^{n/2}}{\Gamma\left(n/2\right)} \cdot \frac{1}{r_0} \tag{A15}$$

and the normalized distribution becomes for all $n \geq 1$

$$p\left(w_1, \ldots, w_n\right) = r_0 \frac{\Gamma\left(n/2\right)}{2\pi^{n/2}} \cdot \frac{1}{\left[w_1^2 + \ldots + w_n^2\right]^{\frac{n+1}{2}}}. \tag{A16}$$

The remaining question concerns of course the choice of r_0. Let $a_1^*, \ldots, a_n^*$ be a point through which the hyperplane is expected to pass. Then Eq. A2 holds for this point. The minimum value of r given this point can be determined by solving the Lagrangian optimization problem

$$\Phi = w_1^2 + w_2^2 + \ldots + w_n^2 + 2\lambda\left(w_1 a_1^* + \ldots + w_n a_n^* + 1\right). \tag{A17}$$

and results in

$$r_{\min}^2 = \frac{1}{\sum_i a_i^{*2}}. \tag{A18}$$

The reciprocal of $r_{\min}$ is therefore the minimal distance of the decision boundary to the origin. A suitable scale for r_0 can therefore be obtained from that point $\vec{a^*}$ which lies farthest away from the origin of the data cloud or by using the radius of the smallest enclosing sphere as guidance. The value of r_0 should be chosen such that decision boundaries approaching this value do not longer vary within the region of interest.

[1] MACKAY, D.J.C. (1992). A practical Bayesian framework for backpropagation networks, in Neural Computation,**4**(3) ,p.448-472.
[2] NEAL, R.M. (1992). Bayesian training of backpropagation networks, Technical report CRG-TR-92-1, Departement of Computer Science, University of Toronto.
[3] BISHOP, C.M. (1995). Neural networks for pattern recognition, Oxford University Press.
[4] LAMPINEN, J. and VEHTARI, A. (2001). Bayesian approach for neural networks, in Neural Networks, **14**, p.257-274.
[5] WILLIAMS, P.M. (1995). Bayesian regularization and pruning using a laplace prior, in Neural Computation, **7**, p.117-143.
[6] WILLIAMS, P.M., Matrix logarithm parametrizations for neural network covariance models, in Neural Networks, **12** (1999), p.299-308.
[7] GOUTTE, C. and HANSEN, L.K. (1997). Regularization with a pruning prior in Neural Networks, **10**(6), p.1053-1059.
[8] JAYNES, E.T. (1983). Prior probabilities in *Papers on Probability, Statistics and Statistical Physics*, edited by R. Rosenkrantz, Reidel, Dordrecht, The Netherlands.
[9] BUCK, B. and MACAULAY, V.A. (1991). *Maximum Entropy in Action*, Oxford University Press.
[10] BOX, G.E.P. and TIAO G.C., *Bayesian Inference in Statistical Analysis*, Addison-Wesley, Reading,MA, 1973.
[11] SIVIA, D. (1996). *Data Analysis: A Bayesian Tutorial*, Oxford University Press.
[12] LEONARD, T. and HSU J.S.J., *Bayesian Methods: An Analysis for Statisticians and Interdisciplinary Researchers*, Cambridge University Press, Cambridge, 1999.
[13] DOSE, V. (2003). Hyperplane priors in *Bayesian Inference and Maximum Entropy Methods in Science and Engineering: 23rd International Workshop*, edited by C. J. Williams, AIP, Melville, NY, p.350-357.
[14] BERGER, E. et al (1997). Approach for the evaluation of speckle deformation measurements by application of the wavelet transformation, in Applied Optics **36** p.7455-7460.
[15] Optimization routine nag_nlp_sol, mark18 from NAG LTD (2003), Oxford,OX2 8DR, UK, http://www.nag.co.uk
[16] PENNY, W.D. and ROBERTS, S.J. (1999). Bayesian neural networks for classification in Neural Networks, **12**, p.877-892.
[17] THODBERG, H.H. (1995). A review of Bayesian neural networks with an application to near infrared spectroscopy in IEEE Transactions on Neural Networks, **7** (1), p.56-72.
[18] BERGER, E. et al (1999). Reconstruction of surfaces from phase-shifting speckle interferometry, in Applied Optics **38**, p.4997-5003.
[19] LANDAU, L.D. and LIFSCHITZ, E.M (1962). Lehrbuch der Theoretischen Physik I, Akademie Verlag, Berlin, 1962.
[] For simplicity we address only regularization but the same is also valid for classification.